Cluster Computing (2021) 24:2479–2494
https://doi.org/10.1007/s10586-021-03278-4

Security-SLA-guaranteed service function chain deployment in cloud-fog computing networks

Dongcheng Zhao[1] · Long Luo[1] · Hongfang Yu[1,2] · Victor Chang[3] · Rajkumar Buyya[4] · Gang Sun[1,5]

Received: 25 August 2020 / Revised: 22 March 2021 / Accepted: 25 March 2021 / Published online: 9 April 2021

Abstract
Network function virtualization (NFV) has gained prominence in next-generation cloud computing, such as the fog-based radio access network, due to their ability to support better QoS in network service provision. However, most of the current service function chain (SFC) deployment researches do not consider the Security-Service-Level-Agreement (SSLA) in the deployment solution. Therefore, in this work, we introduce the SSLA into SFC deployment to defend attacks. Firstly, we formulate the SSLA guaranteed SFC deployment problem by using linear programming. Then, we propose the Maximal-security SFC deployment algorithm (MS) to maximize the security of the SFC deployment. However, the MS algorithm results in a high deployment cost. To reduce the deployment cost, we propose the Minimal-cost and SSLA-guaranteed SFC deployment algorithm (MCSG) to minimize the deployment while satisfying the SSLA. In order to reduce the blocking ratio caused by MCSG, the Minimal-cost and SSLA-guaranteed SFC deployment algorithm with feedback adjustment (MCSG-FA) is proposed. Finally, we evaluate our proposed algorithms through simulations. The simulation results show that the blocking ratio and the deployment cost of our algorithms are better than that of the existing algorithm when meeting the SSLAs.

Keywords Service function chain · Network function virtualization · Deployment · Security · Fog-cloud computing

1 Introduction

Virtualization is a key technology to improve the network flexibility [1–3]. As the development of network, the network function virtualization (NFV) technology has been proposed to transfer the traditional network function to the virtual network function (VNF) to improve the network resource utilization [4]. Multiple VNFs consist of service function chains (SFCs) to guarantee user's service strategy, and the SFC requests are deployed into the cloud network to provide services [5, 6].

With more and more users to use cloud network, the network delay and congestion are becoming more and more serious in the centralized cloud computing.

✉ Gang Sun
gangsun@uestc.edu.cn

Dongcheng Zhao
zhaodc11@gmail.com

Long Luo
longluo.uestc@gmail.com

Hongfang Yu
yuhf@uestc.edu.cn

Victor Chang
victorchang.research111@gmail.com

Rajkumar Buyya
rbuyya@unimelb.edu.au

[1] Key Lab of Optical Fiber Sensing and Communications (Ministry of Education), University of Electronic Science and Technology of China, Chengdu, China

[2] Peng Cheng Laboratory, Shenzhen, China

[3] School of Computing, Engineering and Digital Technologies, Teesside University, Middlesbrough, UK

[4] Cloud Computing and Distributed Systems (CLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, Parkville, Australia

[5] Agile and Intelligent Computing Key Laboratory of Sichuan Province, Chengdu, China

Additionally, the security of service is also being challenged. In order to solve these challenges, distributed fog computing is proposed to extend and supplement the cloud computing [7]. Recently, the fog-based radio access network is becoming a new research hotspot [8]. In fog radio access network (FRAN), there are some fog nodes which similar to cloud nodes. These fog nodes which provide services for users can be virtualized, thereby improving the flexibility of the networks [9]. However, the resource capacity of the fog node is usually less than that of the cloud node. Thus, joint use of cloud-fog computing can decrease network congestion and delay and thus provide high quality services for mobile users.

Currently, there are many researches focus on SFC deployment in cloud computing [10–13]. For example, in [13], Ricard Vilalta et al. conducted a software defined network (SDN)/NFV deployment experiment for 5G services based on cloud-fog computing to optimize the VNF deployment while satisfying the constraints (e.g., latency). With the increase of SFC requests, how to ensure the security-service-level -agreement (SSLA) of services has become a big challenge. In order to guarantee the SSLAs of services when the services are attacked, there are some studies on the security of NFV. For example, Mahdi Daghmehchi Firoozjaei et al. classified security threats and proposed the possible solutions from the architectural layer [14]. These studies proposed some security architectures of NFV to ensure the SSLAs of services from the architecture layer but still cannot defend all attacks. Furthermore, these studies did not consider utilizing the federated environment of the cloud-fog network to provide more secure services for mobile users. In [15], the authors proposed the security-on-demand services for ATM networks, i.e., different user requests have different SSLA requirements. Then, the user request is deployed according to the user’s SSLA requirement. Besides, in [16, 17], the authors discussed how to manage and meet users’ SSLA demands. In commercial applications, some companies (e.g., Huawei) provide servers with different security levels to users to deploy service requests with different SSLA requirements, and the higher security level of the server, the higher charge.

To solve the challenge in management of SSLA in SFCs, in this paper, we consider that each service node and each physical link has a security possibility that can defend attacks. Therefore, we study the SFC placement problem in the federated environment of the cloud-fog network to meet the SSLA requirements of users when each service node and each physical link has a fixed security level. The main contributions of this paper are as follows.

- To resolve the SFC deployment problem with the SSLA requirements, we first formulate the SFC deployment problem with the SSLA requirements by using linear programming.
- To satisfy the security requirements of SFC requests, when the security of each service node and each physical link is given, we propose a maximal-security SFCs deployment algorithm (MS).
- To further reduce the deployment cost, we present a minimal-cost and SSLA-guaranteed SFC deployment algorithm (MCSG).
- Furthermore, to both reduce the deployment cost and blocking ratio, we propose a minimal-cost and SSLA-guaranteed SFCs deployment algorithm with feedback adjustment (MCSG-FA).

The resting of this paper is arranged as follows. Section 2 introduces related work. Section 3 models the studied SFC deployment problem. Section 4 presents the proposed SFC deployment algorithms. We evaluate our proposed algorithms in Sect. 5. Finally, Sect. 6 concludes this work.

2 Related work

2.1 VNF deployment in cloud computing

NFV has become a key technology for improving the network flexibility, which is proposed to transfer the traditional network functions to the VNFs. Multiple VNFs consist of SFC in a specific order, and these SFCs are deployed into the cloud network for providing services to users. There are many researches about SFC deployment in cloud computing [10, 18–23].

In [10], the author studied the problem of traffic-aware and energy-efficient SFC deployment and designed a sampling -based Markov approximation and matching-theoretic algorithm (SAMA) to deploy SFCs into the cloud network for minimizing the deployment cost, but the authors did not consider the SFC deployment in fog computing. In [18], the author researched the shared pipeline problem in the environment of NFV. The study transmitted a plurality of data packets by utilizing the shared pipeline for reducing the core computing resources requirement, but it resulted in increasing the length of the pipeline. In order to solve the problem, the authors presented two heuristic algorithms to balance between the core computing resources requirement and the pipeline length for reducing network latency and resource consumption. To reduce the total VNF delay (including the processing delay and the transmission delay), Long Qu et al. [19] studied the optimal scheduling problem of VNFs. They proposed a heuristic algorithm for optimizing the scheduling of VNFs based on the genetic algorithm can lower the total scheduling time by up to 20%.

The authors in [20] studied the overhead caused by virtual network functions and proposed a new framework based on the programmable software and hardware to obtain the flexibility and the high performance of NFV, and also presented a performance-aware VNF deployment algorithm. To ensure the quality of service (QoS) and reduce the energy consumption of servers, in [21], the authors studied the VNF migration to adjust the workloads of the servers dynamically and presented corresponding heuristic algorithms to reduce the total cost.

In [22], Marcelo Caggiani Luizelli et al. researched the efficient deployment of large-scale VNFs and virtual links, and presented a heuristic algorithm to solve this problem to optimize the virtual network function deployment and reduce resource costs while meeting the network traffic demand. To minimize the placement cost, the authors in [23] researched the optimal placement of the service function chains and put forward a graph algorithm based on a matrix and multi-stage optimization to achieve the goal of reducing costs.

These researches [18–23] studied the problem of deploying SFCs in cloud computing, but they did not consider the NFV security and utilization of the federated environment of the cloud-fog network to provide better services.

2.2 NFV and fog computing

More and more users are accessing the cloud network, centralized cloud computing is facing a big challenge. Thus, the distributed fog computing is proposed to extend and supplement the centralized cloud computing to solve these challenges [24–27]. There are many studies about fog computing in recent years [28–34].

In [28], the authors introduced fog computing into the Internet of things. They put forward a fog computing based model for the problem of face identification, for improving the processing efficiency and reducing the network delay. Kai Liang et al. introduced fog computing into the radio access networks, which combining virtualization and SDN to slice the resources of radio access networks to improve the flexibility of radio access networks [29]. In [30], the authors studied a fog computing-based model of Internet access networks by utilizing virtual machines to host the business of fog network for reducing network latency and improving user experience.

The authors in [31] studied the problem of fog computing access control and proposed a channel encryption and decryption model to improve the security of fog computing to defend network attacks. In [32], Seongjin Park et al. studied the vehicular network's connection problem by utilizing fog computing. In this research, they employed fog node to collect information on the mobile vehicle to realize the corresponding vehicle service to achieve quick connection recovery when a failure occurs. The authors in [33] took into account the problem that fog computing resources cannot satisfy the requirements of vehicle users in the peak hour. The fog vehicular computing concept was proposed to balance the needs of vehicle users and achieve a high utilization of fog computing resources.

In [9], the authors discussed the fusion problem of 5G, cloud-fog computing and NFV, and then put forward a fusion and open architecture to provide the continuous management from cloud computing to fog computing. To meet the performance requirements of 5G services, Ricard Vilalta et al. put forward an NFV architecture for the federated environment of cloud-fog network to provide performance assurance for 5G services [34].

In [28–33], fog computing was widely applied to the Internet of things, radio access network and vehicular network and provided network services to users, but these studies did not research NFV. The research projects [9, 13, 34] studied the NFV problem by utilizing the advantages of fog computing. They only proposed the NFV architecture of the fog computing environment, but the NFV deployment algorithm is not proposed, so deploying NFV in the fog network needs to be studied.

2.3 SSLA in cloud/fog computing

With the explosive growth of service requests and virus attacks are more frequent, ensuring the SSLA of services has become a big challenge. Therefore, there are many studies about SSLA in network function virtualization [14, 35–38].

In NFV, due to the sharing of underlying resources and the live migration of VNFs, VNFs are vulnerable to the shared resource misuse attack and the side-channel attack. In order to solve these security threats, the authors in [14] first classified the attacks and then proposed corresponding solutions. Since SSLA of VNF is very important, the authors discussed the security of NFV architecture and the influence of the outsider attacks and insider attacks [35]. However, that work did not give a corresponding solution to defense these attacks. Due to network function virtualization may bring network attacks to services, it is necessary to enhance the service security. Thus, research [36] proposed a security framework to ensure SFC security.

In recent years, distributed denial of service (DDoS) attacks continue to increase. At the same time, the traditional defense methods are not strong enough. To solve this problem, in [37], Bahman Rashidi et al. presented the DDoS defense mechanism to achieve a collaboration network based on "domain-helps-domain" to deal with a lot of DDoS attacks. Although the DDoS defense mechanism can effectively handle many DDoS attacks, it may not be able to deal with other attacks effectively.

In order to lower the impact of server failures on the network services, the authors [38] studied the high availability deployment problem of the service function chain. They presented an SFC deployment algorithm based on service backup. The SFC deployment algorithm can improve the survivability of network services, but it cannot guarantee the SSLA of services.

The studies [14, 35–38] only proposed secure architectures of NFV to guarantee the security of the service from the architecture layer, which cannot defend against all attacks and did not consider utilizing the federated environment of the cloud-fog network to provide more secure services for users. In [39–41], the authors studied the security of fog computing and proposed some architectures and defense mechanisms to guarantee fog computing security. Additionally, the fog radio access network has smaller coverage than the cloud network, which helps deploy defense hardware to defend attacks. Thus, the fog radio access network can provide higher security than the cloud network does. Therefore, the security deployment problem of SFC is worth further research for providing more secure services for users. The authors in [42] studied the security-aware virtual network mapping problem in the cloud environment, but they did not consider the fog computing environment. It is not appropriate to take the least security value of all links in a path as the security value of the path when the authors computed the security of the embedding path in that research.

3 Problem description and modelling

3.1 Problem description

In this research, the physical network and SFC deployment requests of mobile users with SSLA requirements are given. Similar with Ref. [42], we consider each physical node and each physical link with a security possibility that can defend attacks. Under the given security level of each physical node and link, we propose SFC deployment algorithms for reducing the blocking ratio and the deployment cost while meeting the SSLAs of mobile users.

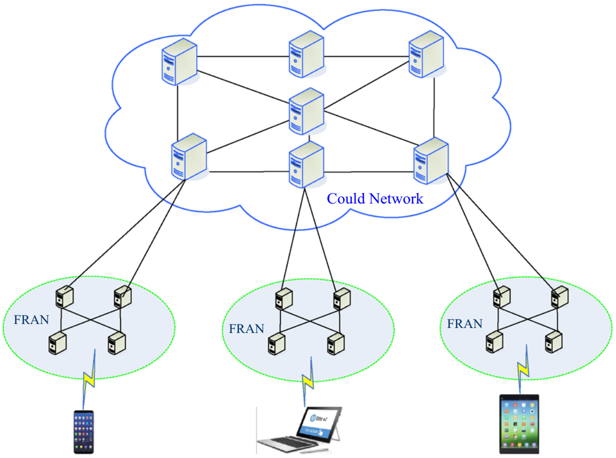

Fig. 1 An example of physical network

3.2 Physical network

In this work, the physical network includes two parts: the centralized cloud network and multiple distributed FRANs. An example of physical network is shown in Fig. 1. We indicate the physical network as $G_P = (N_P, E_P)$. Where, $N_P = \{n_1, n_2, \ldots, n_{|NP|}\}$ indicates the set of the physical nodes in the physical network, $|NP|$ indicates the number of the physical nodes. $E_P = \{l_1, l_2, \ldots, l_{|EP|}\}$ denotes the set of the physical links. $|EP|$ indicates the number of physical links.

Resource constraints of physical network: we define the physical network resource constraints as $RC = (C_{NP}, C_{EP}, S_{NP}, S_{EP}, L_{NP})$.

Resource attributes of physical node: we use C_{NP} to represent the resource attributes set of the physical nodes, which consist of the unit cost $p(n_i)$ of the resource of physical node and the capacity $c(n_i)$ of the computing resource of physical node.

Resource attributes of physical link: C_{EP} denotes the resource attributes set of physical links, which consists of the unit cost $p(l_i)$ of the resource of physical link and the resource capacity $b(l_i)$ of physical link.

Security level of physical node: It is a numerical concept of the security level of the physical node [42]. We use $S_{NP} = \{s(n_1), s(n_2), \ldots, s(n_{|NP|})\}$ to denote the security level set of physical nodes.

Security level of physical link: It's a numerical concept of the security level of physical link [42]. We use $S_{EP} = \{s(l_1), s(l_2), \ldots, s(l_{|EP|})\}$ to denote the security level set of physical link. While maintaining the security of the physical node/link may need to purchase the specialized hardware or invest in human resources, so the higher the security level, the higher the cost of unit resource of the physical node/link.

Locations of physical nodes: We use the notation L_{NP} = $\{L(n_1), L(n_2),\ldots, L(n_{|NP|})\}$ to denote the set of locations of all physical nodes.

3.3 SFC request

An SFC request of mobile user includes two parts: VNFs of cloud network and VNFs of fog radio access network. In traditional access network, the network functions of radio access network are realized by dedicated hardware, which performs the network functions of user's traffic billing, user management and IP address allocation to access to an external network. In order to facilitate the management of users, these network functions are all implemented in access network. In NFV environment, these network functions implemented through virtualization also need to be implemented in the corresponding access network to manage users effectively. Figure 2 shows an example of an SFC request. Similar with the physical network, we can denote an SFC deployment request as $G_F = (N_F, E_F)$. Where $N_F = \{vf_1, vf_2,\ldots, vf_{|NF|}\}$ indicates the set of the VNFs, $|NF|$ is the number of the VNFs in the SFC request. $E_F= \{e_1, e_2,\ldots, e_{|EF|}\}$ denotes the set of links of the SFC request. $|EF|$ indicates the number of links in the SFC request.

Deployment constraints: We define the deployment constraints of the SFC deployment request as $DC = (C_{NF}, C_{EF}, SR, L_{NF}, L_T, L_U)$.

Resource constraints of VNFs: We define $C_{NF}= \{\varepsilon(vf_1), \varepsilon(vf_2),\ldots, \varepsilon(vf_{|NF|})\}$ as the computing resource constraint set of all VNFs.

Resource constraints of SFC links: $C_{EF} = \{\varepsilon(e_1), \varepsilon(e_2),\ldots, \varepsilon(e_{|EF|})\}$ represents the bandwidth resource constraint set of SFC links.

SSLA requirement of an SFC request: We define SR as the overall SSLA constraints of an SFC request.

Location constraints of VNFs, service terminal and mobile user: We use $LC_{NF}= \{LC(vf_1), LC(vf_2),\ldots, LC(vf_{|NF|})\}$ to denote the set of location constraints of all VNFs. VNFs of the cloud network can only be deployed into cloud network, and VNFs of the fog radio access network can only be mapped into FRAN in which the mobile user is located. L_T represents the location of service terminal. L_U denotes the location of mobile user.

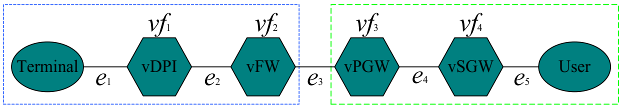

Fig. 2 An example of SFC request

3.4 Modelling for SFC deployment

For provisioning an SFC request, we have to effectively deploying the VNFs and links. The deployment for VNFs can be formulated as:

$$DS : (N_F, C_{NF}) \longrightarrow DS(N_{P1}, C_{NP1}),$$
$$DS(vf_i) \in N_{P1}, \forall vf_i \in N_F,$$
$$A(DS(vf_i)) \geq \varepsilon(vf_i), \forall vf_i \in N_F,$$
$$Z(LC(vf_i), q) \in \{0,1\}, \forall vf_i \in N_F, \forall q \in \{0,1,\ldots,Q\},$$
$$L(DS(vf_i)) \in \{0,1,2,\ldots,Y\}, \forall vf_i \in N_F,$$
$$Z(LC(vf_i), L(DS(vf_i))) = 1, \forall vf_i \in N_F,$$

where $DS = (DS_N, DS_E)$ denotes the deployment solution of the SFC request; $DS_N= \{DS(vf_1), DS(vf_2),\ldots, DS(vf_{|NF|})\}$ denotes the set of placement solutions of all VNFs; $DS_E= \{DS(e_1), DS(e_2),\ldots, DS(e_{|EF|})\}$ denotes the set of placement solutions of all SFC links. $N_{P1} \subset N_P$ indicates a subset of the physical nodes for hosting the VNFs; $C_{NP1} \subset C_{NP}$ describes the node resources allocated to the SFC request; $DS(vf_i)$ denotes a physical node for hosting the *i-th* VNF vf_i; $A(DS(vf_i))$ denotes the available node resources of the physical node $DS(vf_i)$; $q\in 0, 1,\ldots,Q$ indicates the number of the network areas; $Z(LC(vf_i), q)\in\{0,1\}$ indicates a binary variable, when $Z(LC(vf_i),q) = 1$ denotes that VNF vf_i can be deployed into the network area, otherwise $Z(LC(vf_i), q) = 0$. $L(DS(vf_i))$ describes the number of network areas of physical node $DS(vf_i)$. $Z(LC(vf_i), L(DS(vf_i))) = 1$ describes that physical node $DS(vf_i)$ satisfies the location constraint of VNF vf_i, otherwise $Z(LC(vf_i), L(DS(vf_i))) = 0$.

In this work, we deploy the SFC links when we deploy the VNFs, and we formulate the deployment of SFC links as:

$$DS : (E_F, C_{EF}) \longrightarrow DS(P1, C_{EP1}),$$
$$DS(e_i) = p_{e_i},\ \forall e_i \in E_F, \exists p_{e_i} \in P1,$$
$$B(p_{e_i}) = \min_{l_j \in p_{e_i}}\{b(l_j)\} \geq \varepsilon(e_i),\ \exists p_{e_i} \in P1,$$

wherein $P1$ indicates a subset of the physical paths; C_{EP1} denotes the resources of physical links allocated for the SFC request; p_{ei} and $DS(e_i)$ show a physical path for hosting the SFC link e_i; $B(p_{ei})$ used to denote the available bandwidth resource of physical path p_{ei}.

$$\begin{aligned} &TSecurity(DS) \\ &= \{\prod_{i=1}^{|NF|} VNFSecurity(DS(vf_i))\}\{\prod_{i=1}^{|EF|} PathSecurity(DS(e_i))\} \\ &=\{\prod_{i=1}^{|NF|} s(DS(vf_i))\}\{\prod_{i=1}^{|EF|} \prod_{l_k \in DS(e_i)} s(l_k)\} \end{aligned} \tag{1}$$

For a given deployment solution *DS* of an SFC, the security can be computed through Eq. (1).

To satisfy the SSLA requirements of an SFC request, we first consider maximizing the security of the SFC and model the maximal-security deployment as the following linear programming problem (2).

$$
\begin{aligned}
&\max(\{\prod_{i=1}^{|NF|} S(DS(vf_i))\}\{\prod_{i=1}^{|EF|} \prod_{l_k \in DS(e_i)} s(l_k)\})\\
&s.t.\\
&A(DS(vf_i)) \geq \varepsilon(vf_i), \forall vf_i \in N_F\\
&Z(LC(vf_i), q) \in \{0,1\}, \forall vf_i \in N_F, \forall q \in \{0,1,\ldots,Q\}\\
&L(DS(vf_i)) \in \{0,1,2,\ldots,Q\}, \forall vf_i \in N_F\\
&Z(LC(vf_i), L(DS(vf_i))) = 1, \forall vf_i \in N_F\\
&DS(e_i) = p_{e_i}, \quad \forall e_i \in E_F, \exists p_{e_i} \in P1\\
&B(p_{e_i}) = \min_{l_j \in p_{e_i}}\{b(l_j)\} \geq \varepsilon(e_i), \quad \exists p_{e_i} \in P1
\end{aligned}
\tag{2}
$$

Suppose the security of the SFC deployment solved by the linear programming (2) does not meet the SSLA requirement of the SFC request. In that case, the SFC request will be rejected because the current network cannot provide a deployment with sufficient security. Since the security level is higher, the unit cost of the physical node/link is the higher, and the maximal-security implementation may lead to a higher deployment cost. To minimize the total deployment cost when the SSLA requirements of the SFC deployment request is guaranteed, we model the minimal-cost and SSLA-guaranteed SFC provision as the linear programming problem (3).

$$
\begin{aligned}
&\min(\sum_{vf_i \in N_F} P(DS(vf_i))\varepsilon(vf_i) + \sum_{e_i \in E_F} \sum_{l_j \in p_{e_i}} P(l_j)\varepsilon(e_i))\\
&s.t.\\
&\{\prod_{i=1}^{|NF|} S(DS(vf_i))\}\{\prod_{i=1}^{|EF|} \prod_{l_k \in DS(e_i)} s(l_k)\} \geq SR\\
&A(DS(vf_i)) \geq \varepsilon(vf_i), \forall vf_i \in N_F\\
&Z(LC(vf_i), q) \in \{0,1\}, \forall vf_i \in N_F, \forall q \in \{0,1,\ldots,Q\}\\
&L(DS(vf_i)) \in \{0,1,2,\ldots,Q\}, \forall vf_i \in N_F\\
&Z(LC(vf_i), L(DS(vf_i))) = 1, \forall vf_i \in N_F\\
&DS(e_i) = p_{e_i}, \quad \forall e_i \in E_F, \exists p_{e_i} \in P1\\
&B(p_{e_i}) = \min_{l_j \in p_{e_i}}\{b(l_j)\} \geq \varepsilon(e_i), \quad \exists p_{e_i} \in P1
\end{aligned}
\tag{3}
$$

4 Algorithm design

Due to the SFC deployment with the SSLA requirement is an NP-hard problem, the linear programming cannot gain a deployment solution *DS* in polynomial time. To solve this problem, we put forward a security-aware SFC deployment (SASFCD) algorithm. The SASFCD algorithm can be used in three different scenarios by calling different sub-algorithms, i.e., maximal-security SFC deployment algorithm (MS), minimal-cost and SSLA-guaranteed SFCs deployment algorithm (MCSG) and minimal-cost and SSLA-guaranteed SFCs deployment algorithm with feedback adjustment (MCSG-FA). We assume that the SFC deployment requests follow the Poisson process to arrive dynamically. All arrived SFC requests are stored in the queue *ArrivalSFC*. The finished SFC requests are stored in the set *FinishedSFC*. Each SFC deployment request in the queue *ArrivalSFC* is deployed one by one. We use SFC_{blo} to indicate the set of blocked SFC requests owing to the resource limitation. *Algorithm* 1 shows the SASFCD algorithm

```
Algorithm 1: SASFCD algorithm
Input:1. Physical network $G_P$ = ($N_P$, $E_P$) and resource
         constraints $RC = (C_{NP}, C_{EP}, S_{NP}, S_{EP}, LC_{NP})$.
      2. SFC request queue ArrivalSFC.
Output: Total deployment cost TCost and the blocked SFC
        set $SFC_{blo}$.
Initialization: set TCost=0 and $SFC_{blo}=\emptyset$;
while $ArrivalSFC \neq \emptyset$, do
    Updating resources according to the set FinishedSFC.
    Call MS, MCSG or MCSG-FA for deploying the first
    SFC request $SFC_1$ in ArrivalSFC.
    if the deployment solution DS for $SFC_1$, $DS \neq \emptyset$, then
        updating TCost and the physical network.
    else
        $SFC_{blo} = SFC_{blo} \cup \{SFC_1\}$.
    end if
    $ArrivalSFC = ArrivalSFC \backslash \{SFC_1\}$.
end while
return TCost and $SFC_{blo}$.
```

.

The MS algorithm is used to maximize the security of the SFC deployment solution. In the MS algorithm, we use the maximal-security strategy as a guide strategy for deploying VNF into the most secure physical node and finding the most secure path to maximize the security of the placement solution of SFC, thus get the most secure deployment solution. When we deploy VNF vf_i into the physical node n_j, and find the most secure path p_{ei}, we can find the most secure path $p^{i+1}(n_j, L_U)$ from the current physical node n_j to the user. The aim is to optimize the deployment solution and improve the blocking ratio to ensure the security of the deployment solution. If the security of the SFC deployment solved by the MS algorithm cannot meet the SSLA requirement of the SFC request, the SFC deployment request will be rejected.

The security of VNF vf_i deployed into the physical node n_j, *VNFSecurity*(vf_i→ n_j), that is defined in Eq. (4).

$$VNFSecurity(vf_i \rightarrow n_j) = s(n_j) \tag{4}$$

The deployment cost of VNF vf_i deployed into the physical node n_j, *VNFCost* (vf_i→ n_j), that is defined as in Eq. (5).

$$VNFCost(vf_i \rightarrow n_j) = \varepsilon(vf_i)p(n_j) \tag{5}$$

The security of the physical path p_{ei} for hosting SFC link e_i, *PathSecurity*(p_{ei}), is denoted as in Eq. (6).

$$PathSecurity(p_{e_i}) = \prod_{l_k \in p_{e_i}} s(l_k) \tag{6}$$

The security of the physical path $p^{i+1}(n_j, L_U)$ for hosting link (n_j, L_U), *PathSecurity*($p^{i+1}(n_j, L_U)$), is denoted as in Eq. (7). The physical path $p^{i+1}(n_j, L_U)$ must meet the link resource requirements of SFC link e_{i+1}.

$$PathSecurity(p^{i+1}(n_j, L_U)) = \prod_{l_k \in p^{i+1}(n_j, L_U)} s(l_k) \tag{7}$$

The deployment cost of physical path p_{ei} for hosting SFC link e_i, *PathCost*(p_{ei}), can be computed in Eq. (8).

$$PathCost(p_{e_i}) = \sum_{l_k \in p_{e_i}} p(l_k)\varepsilon(e_i) \tag{8}$$

The security of VNF vf_i deployed into the physical node n_j, *TSecurity*(vf_i→ n_j), that is represented as in Eq. (9).

$$\begin{aligned} &TSecurity(vf_i \rightarrow n_j) \\ &= VNFSecurity(vf_i \rightarrow n_j) \times PathSecurity(p_{e_i}) \\ &\times PathSecurity(p^{i+1}(n_j, L_U)) \\ &= s(n_j)\{\prod_{l_k \in p_{e_i}} s(l_k)\}\{\prod_{l_k \in p^{i+1}(n_j, L_U)} s(l_k)\} \end{aligned} \tag{9}$$

The deployment cost of VNF vf_i deployed into the physical node n_j, *TCost*(vf_i→ n_j), that is denoted as in Eq. (10).

$$\begin{aligned} TCost(vf_i \rightarrow n_j) &= VNFCost(vf_i \rightarrow n_j) + PathCost(p_{e_i}) \\ &= \varepsilon(vf_i)p(n_j) + \sum_{l_k \in p_{e_i}} \{p(l_k)\varepsilon(e_i)\} \end{aligned} \tag{10}$$

The security of the current deployment solution *DS* in the MS algorithm, *TSecurity*(*DS*)', is defined as in Eq. (11).

$$\begin{aligned} &TSecurity(DS)' \\ &= PathSecurity(p^{i+1}(n_j, L_U)) \\ &\times \prod_{k=1}^{i} \{VNFSecurity(DS(vf_k)) \times PathSecurity(DS(e_i))\} \\ &= \{\prod_{l_k \in p^{i+1}(n_j, L_U)} s(l_k)\}\{\prod_{k=1}^{i} [s(DS(vf_k)) \prod_{l_k \in DS(e_i)} s(l_k)]\} \end{aligned} \tag{11}$$

The total deployment cost of the current deployment solution *DS*, *TCost*(*DS*), can be defined as in Eq. (12).

$$\begin{aligned} &TCost(DS) \\ &= \sum_{i=1}^{|NF|} VNFCost(DS(vf_i)) + \sum_{i=1}^{|EF|} PathCost(DS(e_i)) \\ &= \sum_{i=1}^{|NF|} \varepsilon(vf_i)p(DS(vf_i)) + \sum_{i=1}^{|EF|} \sum_{l_k \in DS(e_i)} p(l_k)\varepsilon(e_i) \end{aligned} \tag{12}$$

***Algorithm* 2:** MS algorithm

```
Input: 1. Physical network $G_P$ = ($N_P$, $E_P$) and resource
          constraints $RC$ = ($C_{NP}$, $C_{EP}$, $S_{NP}$, $S_{EP}$, $L_{NP}$).
       2. SFC request $G_F$ = ($N_F$, $E_F$) and deployment
          constraints $DC$ = ($C_{NF}$, $C_{EF}$, $SR$, $LC_{NF}$, $L_T$, $L_U$).
Output: Deployment solution $DS$ and total deployment cost
        $TCost(DS)$.
for each VNF $vf_i$, $i$=1,2,…, $|NF|$, $vf_i \in N_F$, do
  for each physical node $n_j \in N_P$, do
    if the node $n_j$ meets the location constraint of $vf_i$, then
      Try to deploy $vf_i$ into the physical node $n_j$, calculate
      the security of the deployment solution of $vf_i$
      $VNFSecurity(vf_i \rightarrow n_j)$ and the deployment cost
      $VNFCost(vf_i \rightarrow n_j)$ according to Eqs. (4) and (5).
      Find the secure paths $p_{ei}$ and $p^{i+1}(n_j, L_U)$; Calculate
      the security of the path $p_{ei}$ $PathSecurity(p_{ei})$, the
      security of the path $p^{i+1}(n_j, L_U)$ $PathSecurity(p^{i+1}(n_j,
      L_U))$ and the deployment cost $PathCost(p_{ei})$
      according to Eqs. (6), (7) and (8); Calculate the
      security $TSecurity(vf_i \rightarrow n_j)$ and the total deployment
      cost $TCost(vf_i \rightarrow n_j)$ according to Equations (9) and
      (10).
    end if
  end for
  Find the deployment solution of VNF $vf_i$ with the
  maximal security $TSecurity(vf_i \rightarrow n_j)$, and store the
  deployment solutions of $vf_i$ and $e_i$ in $DS$.
  if the deployment solution cannot be found, then
    Clear $DS$ and let $TCost(DS)$ = 0.
    return $DS$ and $TCost(DS)$.
  end if
  Calculate the security $TSecurity(DS)'$ according to
  Equation (11).
  if $TSecurity(DS)' < SR$, then
    Clear $DS$ and let $TCost(DS)$ = 0.
    return $DS$ and $TCost(DS)$.
  end if
end for
Find the most secure path $p_{e_{|EF|}}$ and store it in $DS$,
calculate the total deployment cost $TCost(DS)$ according
to Equation (12).
return $DS$ and $TCost(DS)$.
```

A higher security of the deployment leads to a higher total deployment cost in the MS algorithm. To minimize the total deployment cost, while satisfying the SSLA requirement of the SFC request, we propose the MCSG algorithm. In MCSG algorithm, we use the minimal-cost and SSLA-guaranteed strategy to deploy VNFs for minimizing the total placement cost of the SFC deployment solution. In MCSG algorithm, similar to the MS algorithm, we will find the most secure path $p^{i+1}(n_j, L_U)$ from the current physical node n_j to user, to reduce the total cost.

$MaxSecurity(p^{i+1}(n_j, L_U))$ denotes the maximal security of the pre-deployment solution of the rest of VNFs that we pre-deploy the rest of VNFs into the most secure physical node on the physical path $p^{i+1}(n_j, L_U)$, it is defined as in Eq. (13). Where, $n_t \epsilon p^{i+1}(n_j, L_U)$ denotes the physical node on the physical path $p^{i+1}(n_j, L_U)$.

$$MaxSecurity(p^{i+1}(n_j, L_U)) = \prod_{k=i+1}^{|NF|} \max\{s(n_t), \forall n_t \in p^{i+1}(n_j, L_U)\} \qquad (13)$$

The deployment cost of physical path $p^{i+1}(n_j, L_U)$ for hosting link (n_j, L_U), $PathCost(p^{i+1}(n_j, L_U))$, can be computed in Eq. (14).

$$PathCost(p^{i+1}(n_j, L_U)) = \sum_{l_k \in p^{i+1}(n_j, L_T)} p(l_k)\varepsilon(e_{i+1}) \qquad (14)$$

The security of the current deployment solution DS in the MCSG algorithm, $TSecurity(DS)$", is denoted as in Eq. (15).

$$\begin{aligned} TSecurity(DS)'' &= PathSecurity(p^{i+1}(n_j, L_U)) \\ &\times MaxSecurity(p^{i+1}(n_j, L_U)) \\ &\times \prod_{k=1}^{i}\{VNFSecurity(DS(vf_k)) \times PathSecurity(DS(e_i))\} \\ &= \{\prod_{l_k \in p^{i+1}(n_j, L_U)} s(l_k)\}\{\prod_{k=i+1}^{|NF|} \max\{s(n_t), \forall n_t \in p^{i+1}(n_j, L_U)\}\} \\ &\{\prod_{k=1}^{i}[s(DS(vf_k)) \prod_{l_k \in DS(e_i)} s(l_k)]\} \end{aligned} \qquad (15)$$

The total deployment cost of VNF vf_i deployed into the physical node n_j in the MCSG or MCSG-FA algorithm, $TCost(vf_i \rightarrow n_j)$", that is denoted as in Eq. (16).

$$\begin{aligned} TCost(vf_i \rightarrow n_j)'' &= VNFCost(vf_i \rightarrow n_j) + PathCost(p_{e_i}) \\ &+ PathCost(p^{i+1}(n_j, L_U)) \\ &= \varepsilon(vf_i)p(n_j) + \sum_{l_k \in p_{e_i}} p(l_k)\varepsilon(e_i) + \sum_{l_k \in p^{i+1}(n_j, L_T)} p(l_k)\varepsilon(e_{i+1}) \end{aligned} \qquad (16)$$

***Algorithm* 3:** MCSG algorithm

```
Input: 1. Physical network $G_P = (N_P, E_P)$ and resource
          constraints $RC = (C_{NP}, C_{EP}, S_{NP}, S_{EP}, L_{NP})$.
       2. SFC request $G_F = (N_F, E_F)$ and deployment
          constraints $DC = (C_{NF}, C_{EF}, SR, LC_{NF}, L_T, L_U)$.
Output: Deployment solution $DS$ and total deployment cost
        $TCost(DS)$.
for each VNF $vf_i$, $i$=1,2,…, $|NF|$, $vf_i \in N_S$, do
    for each physical node $n_j \in N_P$, do
        if the node $n_j$ meets the location constraint of $vf_i$, then
            Try to deploy $vf_i$ into the physical node $n_j$; Calculate
              the security of the deployment solution of $vf_i$
              $VNFSecurity(vf_i \rightarrow n_j)$ and the deployment cost
              $VNFCost(vf_i \rightarrow n_j)$ according to Eqs. (4) and (5).
            Find the most secure paths $p_{ei}$ and $p^{i+1}(n_j, L_U)$;
              Calculate the security of path $p_{ei}$ $PathSecurity(p_{ei})$,
              the security of path $p^{i+1}(n_j, L_U)$ $PathSecurity(p^{i+1}(n_j,
              L_U))$, the maximal security of the pre-deployment of
              the rest of VNFs $MaxSecurity(p^{i+1}(n_j, L_U))$, the
              deployment cost $PathCost(p_{ei})$ and the deployment
              cost $PathCost(p^{i+1}(n_j, L_U))$ according to Equations
              (6), (7), (13), (8) and (14), respectively; Calculate
              the security $TSecurity(DS)''$ and deployment cost
              $TCost(vf_i \rightarrow n_j)''$ according to Eqs. (15) and (16).
        end if
    end for
    Find the deployment solution of VNF $f_i$ with the minimal
      deployment cost $TCost(vf_i \rightarrow n_j)''$ and the total security
      $TSecurity(DS)'' \geq SR$; Store the deployment solutions
      of $vf_i$ and $e_i$ in $DS$.
    if the deployment solution cannot be found, then
        Clear $DS$ and let $TCost(DS) = 0$.
        return $DS$ and $TCost(DS)$.
    end if
end for
Find the most secure path $p_{e_{|EF|}}$ and store it in $DS$;
  Calculate the deployment cost $TCost(DS)$ according to
  Equation (12).
return $DS$ and $TCost(DS)$.
```

MCSG algorithm uses minimal-cost and SSLA-guaranteed strategy for deploying VNF to minimize the total placement cost. However, MCSG algorithm reduces the total placement cost by using the security-guaranteed strategy that will cause an increase in the blocking ratio. Therefore, we propose the MCSG-FA algorithm to improve the blocking ratio. In the MCSG-FA algorithm, we first call the MS algorithm to get an initial deployment solution with maximal security. We try to find a deployment solution with minimal total deployment cost. If we find a new deployment solution, and the security of the new deployment solution meets the SSLA requirement of the SFC requests. The total placement cost of the new placement solution is less than that of the initial deployment solution, and we use the new placement solution to replace the initial one. The MCSG-FA algorithm can improve the total deployment cost and the blocking ratio through the feedback adjustment approach.

The security of current deployment solution *DS*’ in the MCSG-FA algorithm, *TSecurity*(*DS*’), can be computed by Eq. (17).

$$
\begin{aligned}
&TSecurity(DS') \\
&= \prod_{k=1}^{i} \{VNFSecurity(DS'(vf_k)) \times PathSecurity(DS'(e_i))\} \\
&=\{\prod_{k=1}^{i} [s(DS'(vf_k)) \prod_{l_k \in DS'(e_i)} s(l_k)]\}
\end{aligned}
\tag{17}
$$

The total deployment cost of current deployment solution *DS*’ in the MCSG-FA algorithm, *TCost*(*DS*’), can be defined in Eq. (18).

$$
\begin{aligned}
&TCost(DS') \\
&= \sum_{i=1}^{|NF|} VNFCost(DS'(vf_i)) + \sum_{i=1}^{|EF|} PathCost(DS'(e_i)) \\
&= \sum_{i=1}^{|NF|} \varepsilon(vf_i)p(DS'(vf_i)) + \sum_{i=1}^{|EF|} \sum_{l_k \in DS'(e_i)} p(l_k)\varepsilon(e_i)
\end{aligned}
\tag{18}
$$

```
Algorithm 4: MCSG-FA algorithm
Input: 1. Physical network $G_P$ = ($N_P$, $E_P$) and resource
          constraints $RC = (C_{NP}, C_{EP}, S_{NP}, S_{EP}, L_{NP})$.
       2. SFC request $G_F$ = ($N_F$, $E_F$) and deployment
          constraints $DC = (C_{NF}, C_{EF}, SR, LC_{NF}, L_T, L_U)$.
Output: Deployment solution DS and total deployment cost
        TCost(DS).
Call MS algorithm to achieve DS and TCost(DS).
if $DS \neq \emptyset$, then
  for each VNF $vf_i$, $i$=1,2,..., |NF|, $vf_i \in N_F$, do
    for each physical node $n_j \in N_P$, do
      if node $n_j$ meets the location constraint of $vf_i$, then
        Try to deploy $vf_i$ into the physical node $n_j$;
        Calculate the security of deployment solution of
        $vf_i$ $VNFSecurity(vf_i \rightarrow n_j)$ and the deployment cost
        $VNFCost(vf_i \rightarrow n_j)$ according to Eqs. (4) and (5).
        Find the minimal cost paths $p_{ei}$, $p^{i+1}(n_j, L_U)$;
        Calculate security $PathSecurity(p_{ei})$, deployment
        costs $PathCost(p_{ei})$ and $PathCost(p^{i+1}(n_j, L_U))$,
        and total deployment cost $TCost(vf_i \rightarrow n_j)''$
        according to Eqs. (6), (8), (14) and (16).
      end if
    end for
    Find the deployment solution of VNF $vf_i$ with the
    minimal deployment cost $TCost(vf_i \rightarrow n_j)$, and store the
    deployment solutions of $vf_i$ and $e_i$ in $DS'$.
    if the deployment solution cannot be found, then
      Clear $DS'$ and let $TCost(DS') = 0$.
      return DS and TCost(DS).
    end if
    Calculate the security $TSecurity(DS')$ according to
    Equation (17).
    if $TSecurity(DS') < SR$, then
      Clear $DS'$ and let $TCost(DS') = 0$.
      return DS and TCost(DS).
    end if
  end for
  Find the minimal-cost path $p_{e_{|EF|}}$ and store it in $DS'$;
  Calculate the security $TSecurity(DS')$ according to
  Equation (17); Calculate the total deployment cost
  $TCost(DS')$ according to Equation (18).
end if
if find a complete solution $DS'$ and $TSecurity(DS') \geq SR$
and $TCost(DS') < TCost(DS)$, then
  Let $DS = DS'$, $TCost(DS) = TCost(DS')$.
end if
return DS and TCost(DS).
```

Next, we analyze the complexity and security of these proposed algorithms. In MS algorithm, we use the Dijkstra algorithm to find the shortest path, the complexity of the Dijkstra is $O(|NP|^2)$. $|NP|$ denotes the number of physical servers; $|NF|$ denotes the number of VNFs in SFC; c_1, c_2, c_3, c_4, c_5 and c_6 are constants. So, the complexity of the MS algorithm can be evaluated as follows. In line 5: we use the Dijkstra algorithm to find the shortest path, so the complexity is $c_1|NP|^2$. In line 1–13: the complexity is $c_2|NF||NP||NP|^2 = c_2|NF||NP|^3$. In line 19: we use the Dijkstra algorithm to find the shortest path, so the complexity is $c_3|NP|^2$. In line 1–20: the complexity is $c_2|NF||NP|^3 + c_3|NP|^2 = O(|NF||NP|^3)$. Thus, the complexity of MS algorithm is $O(|NF||NP|^3)$.

The complexity of MCSG-FA algorithm can be evaluated as follows. In line 1: we call MS algorithm, the time complexity is $O(|NF||NP|^3)$. In line 7: we employ the Dijkstra algorithm to find the shortest path, so the time complexity is $c_4|NP|^2$. In line 3–9: the time complexity is $c_5|NF||NP||NP|^2 = c_5|NF||NP|^3$. In line 21: we use the Dijkstra algorithm to find the shortest path, so the complexity is $c_6|NP|^2$. In line 2–26: the complexity is $c_5|NF||NP|^3 + c_6|NP|^2 = O(|NF||NP|^3)$. Thus, the complexity of the MCSG-FA algorithm is $O(|NF||NP|^3)$.

Finally, we analyze the security of these algorithms. First, virtualization technology can provide a strong isolation for VNFs to avoid denial of service caused by interference of other VNFs. The strong isolation includes resource isolation, performance isolation, and security isolation, so that each VNF or each virtual link has independence in terms of resources, performance, and security. Furthermore, virtualization technology offers inter- and intra-network QoS provisioning by using a consistent resource controller. Second, we defined the overall security of an SFC deployment in Eq. (1) and assumed each physical node and each physical link to have a security level for defending attacks [42]. Furthermore, in our proposed algorithms, when we deploy an SFC, the MS algorithm uses the maximal-security strategy as a guide strategy for deploying VNFs into the most secure physical node and finding the most reliable paths to ensure the security of SFC deployment. The MCSG algorithm uses the minimal-cost and SSLA-guaranteed strategies for deploying VNFs to ensure the security of the SFC deployment. The MCSG-FA algorithm first calls the MS algorithm to get an initial deployment, then it tries to find a new deployment to minimize the total deployment cost while meeting security requirements. Therefore, we can guarantee the security of an SFC request from two aspects of virtualization technology and the deployment strategy in our proposed algorithms.

5 Performance evaluation and analysis

5.1 Simulation environment

In this work, we consider utilizing the federated environment of the cloud-fog network to provide more secure services for more mobile users. Therefore, the physical network is comprised of cloud network (the USANET network, as shown in Fig. 3) and multiple FRANs (as shown in Fig. 4). In our simulations, there are 15 fog radio access networks that connect to the black nodes numbered 0, 5, 7, 12, 14, 16, 20, 23, 25, 29, 32, 34, 36, 42 and 44.

We presume that the unit cost of the node resource of each physical node in physical network is $\log(1/(1 - s(n_i)))$, the unit cost of the resource of each physical link in physical network is $\log(1/(1 - s(l_i)))$. In our simulations, when we evaluate the total deployment costs and the running time of our proposed algorithms, we assume that the resource capacity constraints of the physical node follow a uniform distribution U(50, 80). The resource capacity constraints of the physical link follow a uniform distribution U(30, 50). When we evaluate the blocking ratios of all algorithms, we assume that the resource capacity of the physical network is unlimited. Without losing generality, we make assumption that: (i) Per unit cost of computing resource is 1 unit and per unit cost of bandwidth resources is 1 unit; (ii) The transmission delay of each core network link is 1 unit.

In our simulations, we assume that 10,000 SFC requests arrive dynamically following a Poisson process when the lengths of SFC requests (i.e., n) varies among 5, 6, 7 and 8, respectively. The resource requirements of VNF and SFC link obey a uniform distribution U(5, 10). We suppose that the location of the service terminal randomly distributed in a physical network node of cloud network, and the location of the mobile user randomly distributed in a physical network node of FRAN. We first find a most secure path p (L_T, L_U) from the service terminal to mobile user. Where, $n_t \in p$ (L_T, L_U) denotes the physical node on the physical path p (L_T, L_U). Then, we set the SSLA requirement of an SFC deployment request according to Eq. (19).

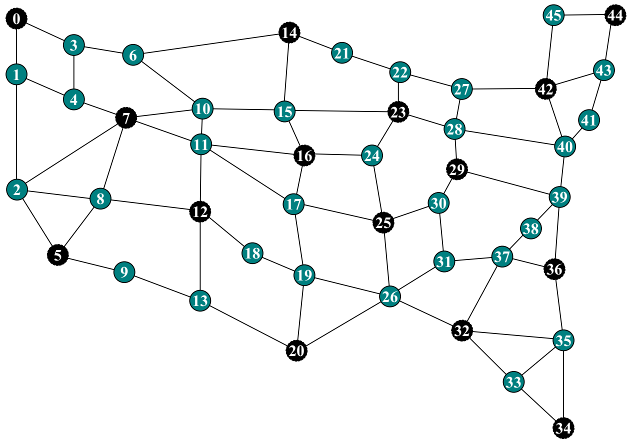

Fig. 3 USANET network

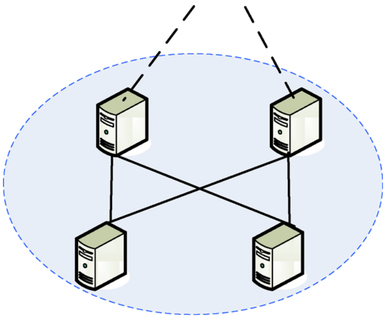

Fig. 4 The topology of a FRAN

$$\begin{aligned} SR &= PathSecurity(p(L_T, L_U)) \\ &\times \prod_{i=1}^{|NF|} \left\{ \begin{array}{r} Average\{s(n_t), \forall n_t \in p(L_T, L_U)\} * 2/3 \\ +\min\{s(n_t), \forall n_t \in p(L_T, L_U)\}/3 \end{array} \right\} \\ &= \{ \prod_{l_k \in p(L_T, L_U)} s(l_k) \} \\ &\times \prod_{i=1}^{|NF|} \left\{ \begin{array}{r} Average\{s(n_t), \forall n_t \in p(L_T, L_U)\} * 2/3 \\ +\min\{s(n_t), \forall n_t \in p(L_T, L_U)\}/3 \end{array} \right\} \end{aligned} \tag{19}$$

In our simulations, we will compare our algorithms with the SAMA algorithm, which is presented in [10] for minimizing the total placement cost.

5.2 Simulation results and analysis

Figure 5 represents the blocking ratios of the MS, MCSG, MCSG-FA and SAMA algorithms, wherein the length of the SFC request (i.e., n) is changed among 5, 6, 7 and 8. From the results, it can be seen that the blocking ratios of our three algorithms are better than that of SAMA algorithm. The SAMA algorithm is proposed to minimize the total deployment cost, so it does not consider the SSLA requirement of SFC request when looking for a deployment solution. Thus, the blocking ratio of the SAMA algorithm is high. Besides, in our three algorithms, when we deploy VNF vf_i into the physical node n_j and find the most secure path p_{ei}, we will find the most secure path $p^{i+1}(n_j, L_U)$ from the current physical node n_j to the user to improve the blocking ratio. The MS algorithm uses the maximal-security strategy to deploy VNF into the most secure physical node and find the most reliable paths to maximize the security of the placement solution of SFC. It can maximize the security of the deployment of SFC and guarantee the success ratio. MCSG algorithm uses the minimal-cost and SSLA-guaranteed strategy for deploying VNF to minimize the total placement cost, but lead an increase in the blocking ratio. So, the blocking ratio of the MCSG algorithm is higher than the blocking ratio of MS algorithm. The MCSG-FA algorithm first calls the MS algorithm to get an initial deployment solution, so that the MCSG-FA

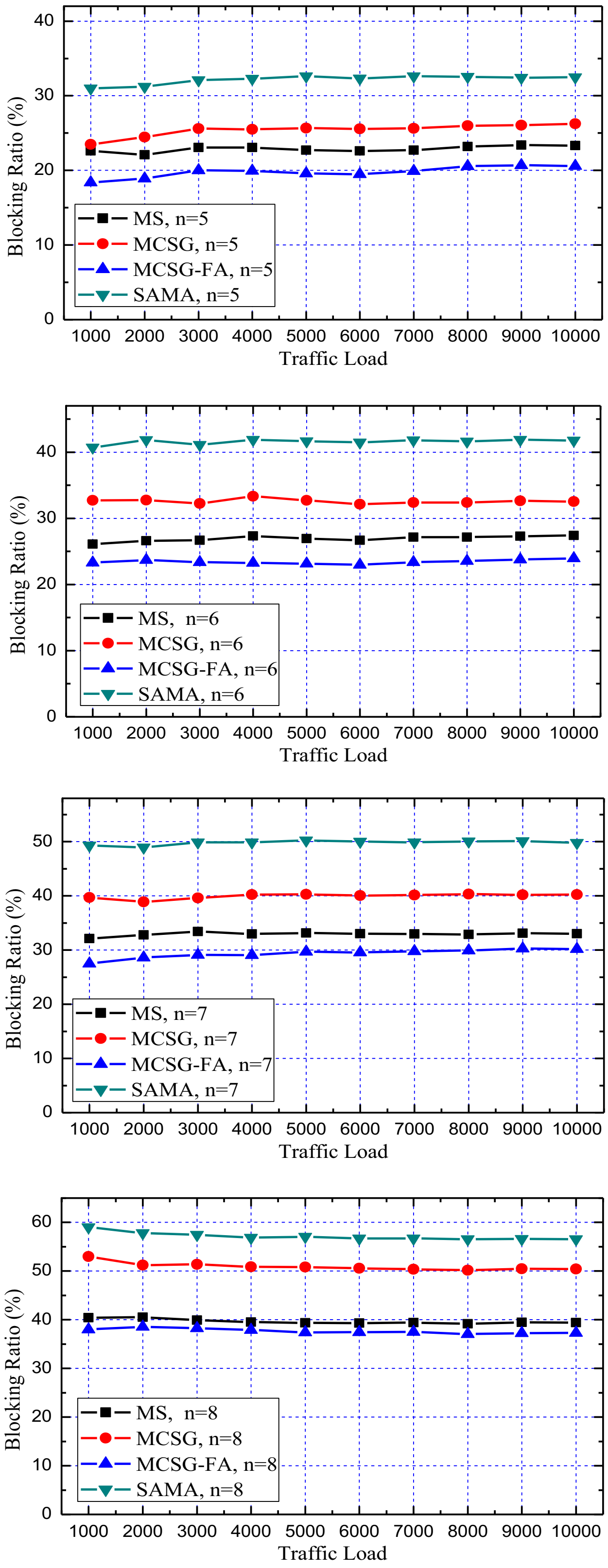

Fig. 5 Simulation results of blocking ratio

algorithm has a similar success ratio to the MS algorithm. Then it tries to find a deployment solution with the minimal deployment cost to replace the initial deployment solution. This can reduce the consumption of network resources to improve the blocking ratio further. Therefore, MCSG-FA algorithm has the lowest blocking ratio.

We compare the total link deployment costs of MS, MCSG, MCSG-FA and SAMA algorithms in Fig. 6, compare the total VNF deployment costs of SAMA algorithm and our three algorithms in Fig. 7, and compare the total SFC deployment costs of four algorithms in Fig. 8. From the results, we can see that the total link deployment cost of MS algorithm is higher than that of SAMA algorithm because MS algorithm pursues the maximal security of the deployment of SFC without considering the deployment cost, whereas SAMA algorithm is designed for minimizing the total placement cost. Hence, the total link deployment cost, the total VNF deployment cost and the total SFC deployment cost of the SAMA algorithm are lower than that of MS algorithm.

Because MCSG algorithm uses the minimal-cost and SSLA-guaranteed strategy for deploying VNFs, it can effectively reduce the link deployment cost, VNF deployment cost and total SFC deployment cost compared to the MS algorithm. The MCSG-FA algorithm first calls MS algorithm to get an initial deployment solution and then tries to find a deployment solution with the minimal total deployment cost to replace the initial deployment solution. It also can lower the link deployment cost, VNF deployment cost and SFC deployment cost compared to the MS algorithm.

Moreover, in MCSG algorithm and MCSG-FA algorithm, when we deploy VNF vf_i and find the most secure path p_{ei}, we will find the most secure or minimal-cost $p^{i+1}(n_j, L_U)$ to reduce the hop of the entire deployment path. Therefore, the MCSG and MCSG-FA algorithms can obtain lower link deployment costs than the SAMA algorithm. We find the path $p^{i+1}(n_j, L_U)$ and use the minimal-cost strategy, and thus deploy VNF into the entire deployment path with lower cost compared to the SAMA algorithm. Hence, MCSG and MCSG-FA algorithms can get the lower total VNF deployment costs than the SAMA algorithm does. So, the total SFC deployment costs of MCSG and the MCSG-FA algorithms are lower than that of the SAMA algorithm.

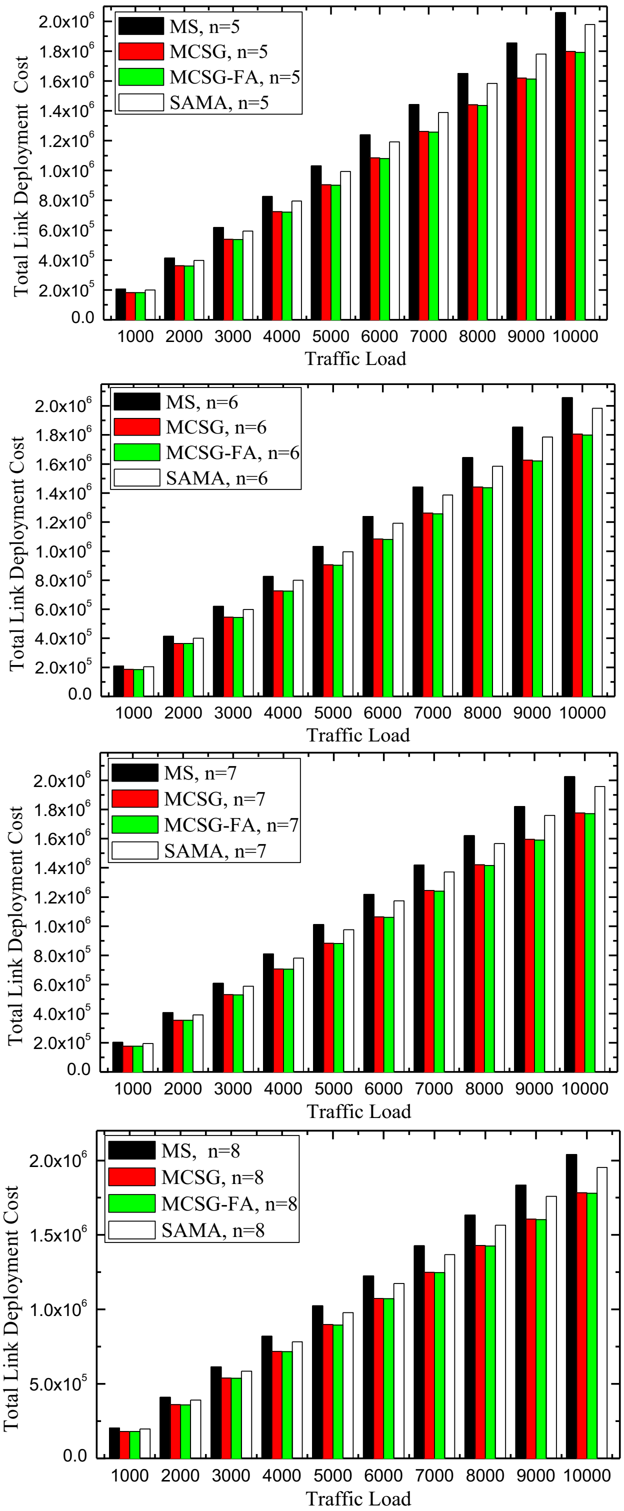

Fig. 6 Simulation results of total link deployment cost

6 Conclusion

In this paper, we investigate the SFC placement problem with SSLA requirement in the federated environment of the cloud-fog networks. To guarantee the security of deployment solution when the security of each physical node and link is given, firstly, we formulate the studied problem as linear programming with SSLA-guaranteed. Then, we propose an algorithm, MS, to maximize the security of the deployment of SFC request. The MS algorithm could maximize the security of SFC deployment, but the total

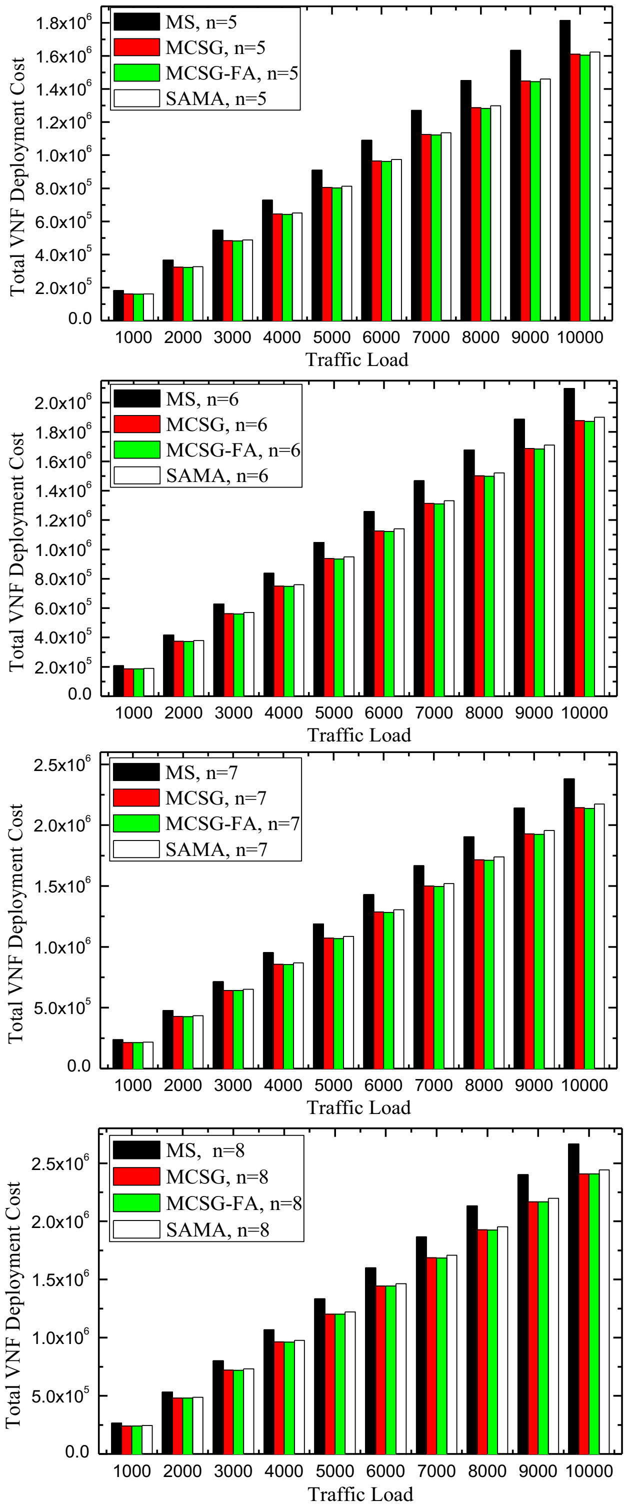

Fig. 7 Simulation results of total VNF deployment cost

deployment cost is high. To reduce the deployment cost, we design an algorithm, MCSG, to minimize the deployment cost and guarantee the SSLA of deployment. Although the MCSG algorithm can reduce the total deployment cost, it results in a higher blocking ratio. To

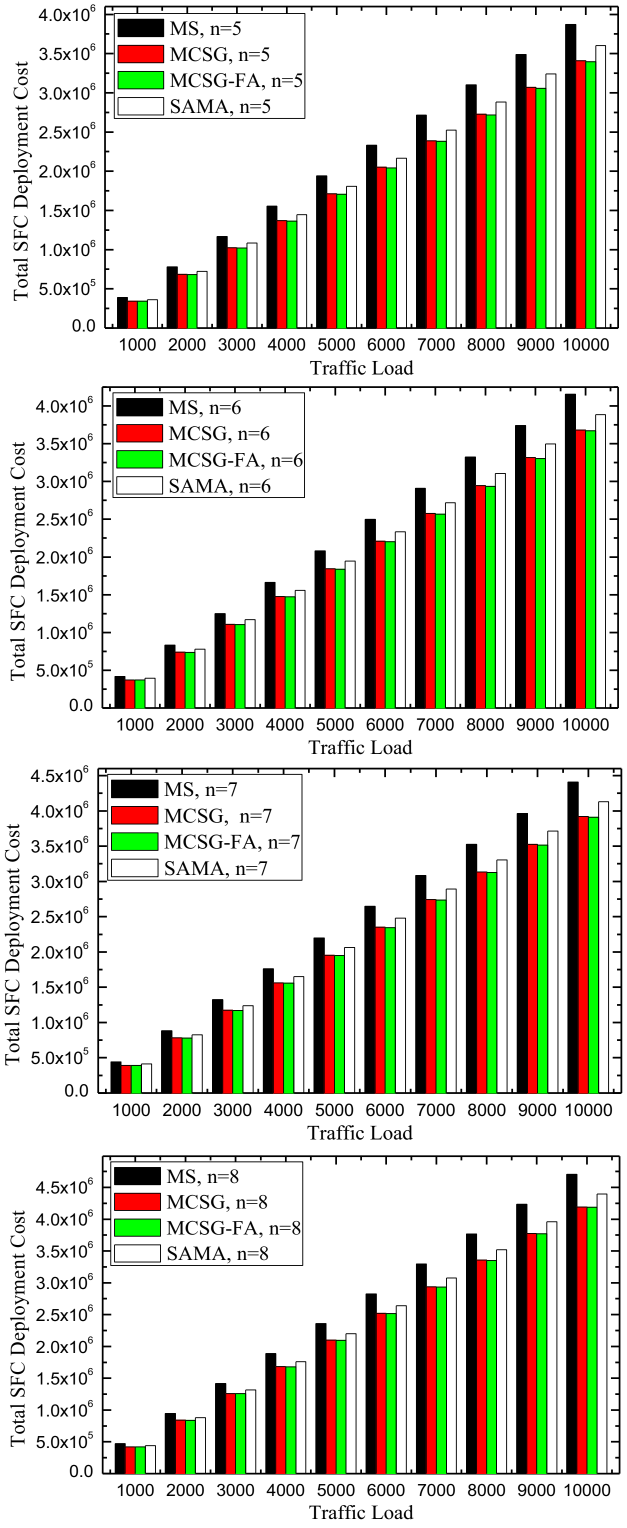

Fig. 8 Simulation results of total SFC deployment cost

both improve the blocking ratio and the total deployment cost, we propose another algorithm, MCSG-FA. We validate our proposed algorithms in the cloud-fog networks. The results reveal that our proposed algorithms have better performance than the existing algorithm in the blocking ratio and the deployment cost.

Our future researches will include the integration between the cloud-fog network and AI-based intelligent systems to make our services more robust, secure and efficient.

Acknowledgements This research was partially supported by the National Key Research and Development Program of China (2019YFB1802800), PCL Future Greater-Bay Area Network Facilities for Large-scale Experiments and Applications (PCL2018KP001), VC Research (VCR 0000086) and VC Research (VCR 0000134) of Prof. Victor Chang.

References

1. Liao, D., Yulong, W., Ziyang, W., Zhu, Z., Zhang, W., Sun, G., Chang, V.: AI-based software-defined virtual network function scheduling with delay optimization. Clust. Comput. **22**(6), 13879–13909 (2019)
2. Sun, G., Liao, D., Zhao, D., Zichuan, X., Hongfang, Yu.: Live migration for multiple correlated virtual machines in cloud-based data centers. IEEE Trans. Serv. Comput. **11**(2), 279–291 (2018)
3. Khairi, S., Raouyane, B., Bellafkih, M.: Novel QoE monitoring and management architecture with eTOM for SDN-based 5G networks. Clust. Comput. **23**, 1–12 (2020)
4. Sun, J., Zhang, Y., Liao, D., Sun, G., Chang, V.: AI-based survivable design for hybrid virtual networks for single regional failures in cloud data centers. Clust. Comput. **22**(5), 12009–12019 (2019)
5. Toosi, A.N., Son, J., Chi, Q., Buyya, R.: ElasticSFC: auto-scaling techniques for elastic service function chaining in network functions virtualization-based clouds. J. Syst. Softw. **152**, 108–119 (2019)
6. Zhao, D., Liao, D., Sun, G., Shizhong, X.: Towards resource-efficient service function chain deployment in cloud-fog computing. IEEE Access **6**(1), 66754–66766 (2018)
7. Zhao, D., Liao, D., Sun, G., Shizhong, X., Chang, V.: On orchestrating service function chains in 5G mobile network. IEEE Access **7**(1), 39402–39416 (2019)
8. Zhao, D., Liao, D., Sun, G., Shizhong, X., Chang, V.: Mobile-aware service function chain migration in cloud-fog computing. Future Gener. Comput. Syst. **96**, 591–604 (2019)
9. Van Lingen, F., Yannuzzi, M., Jain, A., Irons-Mclean, R., Lluch, O., Carrera, D., Pérez, J.L., Gutierrez, A., Montero, D., Martí, J., Masó, R., Rodríguez, J.P.: The unavoidable convergence of NFV, 5G, and fog: a model-driven approach to bridge cloud and edge. IEEE Commun. Mag. **55**(8), 28–35 (2017)
10. Pham, C., Tran, N.H., Ren, S., Saad, W., Hong, C.S.: Traffic-aware and energy-efficient vNF placement for service chaining: joint sampling and matching approach. IEEE Trans. Serv. Comput. **13**(1), 172–185 (2020)
11. Chiang, M., Ha, S., Chih-Lin, I., Risso, F., Zhang, T.: Clarifying fog computing and networking: 10 questions and answers. IEEE Commun. Mag. **55**(4), 18–20 (2017)
12. Jalali, F., Hinton, K., Ayre, R., Alpcan, T., Tucker, R.S.: Fog computing may help to save energy in cloud computing. IEEE J. Sel. Areas Commun. **34**(5), 1728–1739 (2016)
13. Vilalta, R., Mayoral, A., Casellas, R., Martínez, R., Muñoz, R.: Experimental demonstration of distributed multi-tenant cloud/fog and heterogeneous SDN/NFV orchestration for 5G services. In: European Conference on Networks & Communications, pp. 52–56 (2016)
14. Firoozjaei, M.D., Jeong, J., Ko, H., Kim, H.: Security challenges with network functions virtualization. Future Gener. Comput. Syst. **67**, 315–324 (2017)
15. Jerry Schumacher, H.J., Lee, T., Ghosh, S.: A novel, user-level, security-on-demand paradigm for ATM networks: modeling, simulation, and performance analysis. J. Interconnect. Netw. **4**(4), 429–461 (2003)
16. Mthunz, S.N., Benkhelifa, E., Bosakowski, T., Guegan, C.G., Barhamgi, M.: Cloud computing security taxonomy: from an atomistic to a holistic view. Future Gener. Comput. Syst. **107**, 620–644 (2020)
17. Trapero, R., Modic, J., Stopar, M., Taha, A., Suri, N.: A novel approach to manage cloud security SLA incidents. Future Gener. Comput. Syst. **72**, 193–205 (2017)
18. Rottenstreich, O., Keslassy, I., Revah, Y., Kadosh, A.: Minimizing delay in network function virtualization with shared pipelines. IEEE Trans. Parallel Distrib. Syst. **28**(1), 156–169 (2017)
19. Long, Q., Assi, C., Shaban, K.: Delay-aware scheduling and resource optimization with network function virtualization. IEEE Trans. Commun. **64**(9), 3746–3758 (2016)
20. Sun, C., Bi, J., Zheng, Z., Hongxin, H.: HYPER: a hybrid high-performance framework for network function virtualization. IEEE J. Sel. Areas Commun. **35**(11), 2490–2500 (2017)
21. Eramo, V., Miucci, E., Ammar, M., Lavacca, F.G.: An approach for service function chain routing and virtual function network instance migration in network function virtualization architectures. IEEE/ACM Trans. Netw. **25**(4), 2008–2025 (2017)
22. Luizelli, M.C., da Costa Cordeiro, W.L., Buriol, L.S., Gaspary, L.P.: A fix-and-optimize approach for efficient and large scale virtual network function placement and chaining. Comput. Commun. **102**, 67–77 (2017)
23. Khebbache, S., Hadji, M., Zeghlache, D.: Virtualized network functions chaining and routing algorithms. Comput. Netw. **114**, 95–110 (2017)
24. Xiao, Y., Krunz, M.: QoE and power efficiency tradeoff for fog computing networks with fog node cooperation. In: IEEE INFOCOM, pp. 1–9 (2017)
25. Sun, G., Song, L., Hongfang, Yu., Xiaojiang, D., Guizani, M.: A two-tier collection and processing scheme for fog-based mobile crowd sensing in the internet of vehicles. IEEE Internet Things J. **8**(3), 1971–1984 (2021)
26. Song, L., Sun, G., Hongfang, Yu., Xiaojiang, D., Guizani, M.: FBIA: a fog-based identity authentication scheme for privacy preservation in internet of vehicles. IEEE Trans. Veh. Technol. **69**(5), 5403–5415 (2020)
27. Sun, G., Zhang, Y., Hongfang, Yu., Xiaojiang, D., Guizani, M.: Intersection fog-based distributed routing for V2V communication in urban vehicular ad hoc networks. IEEE Trans. Intell. Transp. Syst. **21**(6), 2409–2426 (2020)
28. Pengfei, H., Ning, H., Qiu, T., Zhang, Y., Luo, X.: Fog computing based face identification and resolution scheme in internet of things. IEEE Trans. Ind. Inform. **13**(4), 1910–1920 (2017)
29. Liang, K., Zhao, L., Chu, X., Chen, H.-H.: An integrated architecture for software defined and virtualized radio access networks with fog computing. IEEE Netw. **31**(1), 80–87 (2017)
30. Iotti, N., Picone, M., Cirani, S., Ferrari, G.: Improving quality of experience in future wireless access networks through fog computing. IEEE Internet Comput. **21**(2), 26–33 (2017)
31. Yu, Z., Au, M.H., Xu, Q., Yang, R., Han, J.: Towards leakage-resilient fine-grained access control in fog computing. Future Gener. Comput. Syst. **78**, 763–777 (2018)

32. Park, S., Yoo, Y.: Network intelligence based on network state information for connected vehicles utilizing fog computing. Mob. Inf. Syst. **43**(12), 1420–1427 (2017)
33. Sookhak, M., Richard Yu, F., He, Y., Talebian, H., Safa, N.S., Zhao, N., Khan, M.K., Kumar, N.: Fog vehicular computing: augmentation of fog computing using vehicular cloud computing. IEEE Veh. Technol. Mag. **12**(3), 55–64 (2017)
34. Vilalta, R., Mayoral, A., Casellas, R., Martínez, R., Muñoz, R.: SDN/NFV orchestration of multi-technology and multi-domain networks in cloud/fog architectures for 5G services. In: Optoelectronics & Communications Conference, pp. 1–3 (2016)
35. Aljuhani, A., Alharbi, T.: Virtualized network functions security attacks and vulnerabilities. In: IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC), pp. 1–4 (2017)
36. Fysarakis, K., Petroulakis, N.E., Roos, A., Abbasi, K., Vizarreta, P., Petropoulos, G., Spanoudakis, E.S.G., Askoxylakis, I.: A reactive security framework for operational wind parks using service function chaining. In: IEEE Symposium on Computers and Communications (ISCC), pp. 663–668 (2017)
37. Rashidi, B., Fung, C., Bertino, E.: A collaborative DDoS defence framework using network function virtualization. IEEE Trans. Inf. Forensics Secur. **12**(10), 2483–2497 (2017)
38. Casazza, M., Fouilhoux, P., Bouet, M., Secci, S.: Securing Virtual Network Function Placement with High Availability Guarantees. arXiv, pp. 1–9 (2017)
39. Shirazi, S.N., Gouglidis, A., Farshad, A., Hutchison, D.: The extended cloud: review and analysis of mobile edge computing and fog from a security and resilience perspective. IEEE J. Sel. Areas Commun. **35**(11), 2586–2595 (2017)
40. Roman, R., Lopez, J., Mambo, M.: Mobile edge computing, Fog et al.: a survey and analysis of security threats and challenges. Future Gener. Comput. Syst. **78**, 680–698 (2018)
41. Mukherjee, M., Matam, R., Shu, L., Maglaras, L., Ferrag, M.A., Choudhury, N., Kumar, V.: Security and privacy in fog computing: challenges. IEEE Access **5**, 19293–19304 (2017)
42. Liu, S., Cai, Z., Hong, X., Ming, X.: Towards security-aware virtual network embedding. Comput. Netw. **91**, 151–163 (2015)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Dongcheng Zhao is pursuing his Ph.D. degree in Communication and Information System at University of Electronic Science and Technology of China. His research interests include network function virtualization, cloud computing, fog computing and 5G mobile networks.

Long Luo is a Postdoctoral Researcher at the University of Electronic Science and Technology of China (UESTC) since 2020. She received her BS degree in communication engineering from Xi'an University of Technology in July 2012, MS degree in communication engineering from UESTC in July 2015, and her PhD degree from UESTC in June 2020. Her research interests revolve around software-defined networks, inter\intra-datacenter traffic engineering and data-driven networking.

Hongfang Yu received her B.S. degree in Electrical Engineering in 1996 from Xidian University, her M.S. degree and Ph.D. degree in Communication and Information Engineering in 1999 and 2006 from University of Electronic Science and Technology of China, respectively. From 2009 to 2010, she was a Visiting Scholar at the Department of Computer Science and Engineering, University at Buffalo (SUNY). Her research interests include network survivability and next generation Internet, cloud computing etc.

Victor Chang is currently a Full Professor of Data Science and Information Systems at the School of Computing, Engineering and Digital Technologies, Teesside University, Middlesbrough, UK, since September 2019. He was a Senior Associate Professor, Director of Ph.D. (June 2016–May 2018) and Director of MRes (Sep 2017–Feb 2019) at International Business School Suzhou (IBSS), Xi'an Jiaotong-Liverpool University (XJTLU), Suzhou, China, between June 2016 and August 2019. He was also a very active and contributing key member at Research Institute of Big Data Analytics (RIBDA), XJTLU. He was an Honorary Associate Professor at University of Liverpool. Previously he was a Senior Lecturer at Leeds Beckett University, UK, between Sep 2012 and May 2016. Within 4 years, he completed Ph.D. (CS, Southampton) and PGCert (Higher Education, Fellow, Greenwich) while working for several projects at the same time. Before becoming an academic, he has achieved 97% on average in 27 IT certifications. He won a European Award on Cloud Migration in 2011, IEEE Outstanding Service Award in 2015, best papers in 2012, 2015 and 2018, the 2016 European award. He is a visiting scholar/Ph.D. examiner at several universities, an Editor-in-Chief of IJOCI & OJBD journals, Editor of FGCS, Associate Editor of TII & Information Fusion, founding chair of two international workshops and founding Conference Chair of IoTBDS and COMPLEXIS since Year 2016. He is the founding

Conference Chair for FEMIB since Year 2019. He published 3 books as sole authors and the editor of 2 books on Cloud Computing and related technologies. He gave 18 keynotes at international conferences. He is widely regarded as one of the most active and influential young scientist and expert in IoT/Data Science/Cloud/security/AI/IS, as he has experience to develop 10 different services for multiple disciplines.

Rajkumar Buyya is a Redmond Barry Distinguished Professor and Director of the Cloud Computing and Distributed Systems (CLOUDS) Laboratory at the University of Melbourne, Australia. Dr. Buyya is recognized as a "Web of Science Highly Cited Researcher" in 2016 and 2017 by Thomson Reuters, a Fellow of IEEE, and Scopus Researcher of the Year 2017 with Excellence in Innovative Research Award by Elsevier for his outstanding contributions to Cloud computing.

Gang Sun is a professor of Computer Science at University of Electronic Science and Technology of China (UESTC). His research interests include network virtualization, cloud computing, high performance computing, parallel and distributed systems, ubiquitous/pervasive computing and intelligence and cyber security.